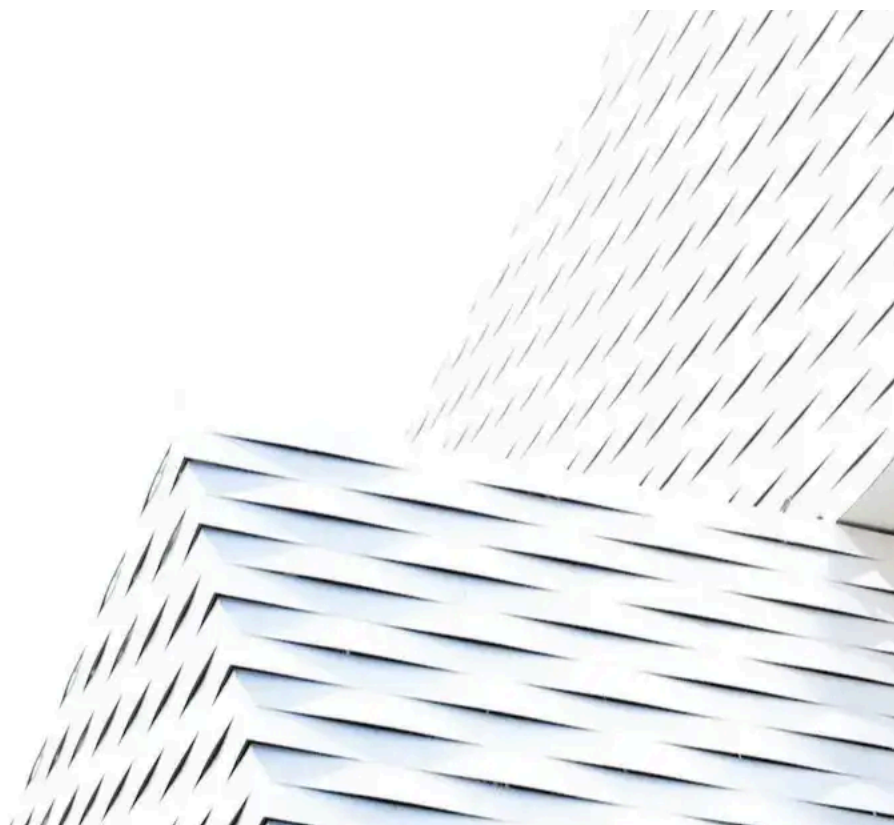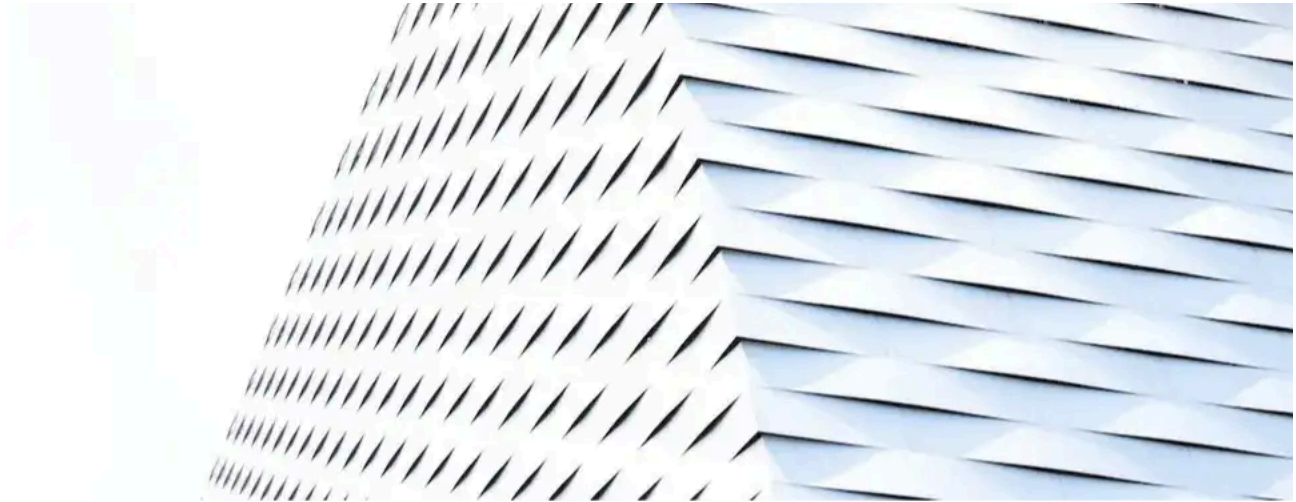.class="img-fluid clearfix"

Insight

Challenges of Rust Library Development

Developing Rust libraries can be a challenging task, especially for beginners. One of the biggest challenges is setting up an efficient project structure and writing all of the necessary boilerplate code. This can be time-consuming and repetitive, and it can take away from the more creative and strategic aspects of library development.

Benefits of Using a Code Generator

Using a code generator can streamline the process of Rust library development by automating boilerplate code generation and other repetitive tasks. This can save developers a significant amount of time and effort, and it can free them up to focus on the more important aspects of library development, such as design, implementation, and testing.

Idea

LibMake: A Code Generator for Rust Libraries

[LibMake](#) ⊞ is a code generator tool designed to quickly help creating high-quality Rust libraries by generating a set of pre-filled and pre-defined templated files. This opinionated boilerplate scaffolding tool aims to greatly reduces development time and minimises repetitive tasks, allowing you to focus on your business logic while enforcing standards, best practices, consistency, and providing style guides for your library.

LibMake is flexible and extensible, so it can be used to create libraries of any size or complexity. It also supports a variety of configuration options, so developers can customise it to their specific needs.

Example of Using LibMake

To use LibMake, developers simply need to run the following command:

```
libmake \
    --author "John Smith" \
    --build "build.rs" \
    --categories "['category 1', 'category 2', 'category 3']" \
    --description "A Rust library for doing cool things" \
    --documentation "https://docs.rs/my_library" \
    --edition "2021" \
    --email "john.smith@example.com" \
    --homepage "https://my_library.rs" \
    --keywords "['rust', 'library', 'cool']" \
    --license "MIT" \
    --name "my_library" \
    --output "my_library" \
    --readme "README.md" \
    --repository "https://github.com/example/my_library" \
    --rustversion "1.69.0" \
```

```
--version "0.1.0" \
--website "https://example.com/john-smith"
```

This will create a new directory for the library, and LibMake will generate the necessary boilerplate code and documentation structure. Developers can then add their own code to the library and start developing.

Impact

Reduced Development Time and Effort

LibMake reduces the time and effort required to develop Rust libraries by automating code generation and other tasks. This saves developers time and effort. They can focus on the important parts, like design, implementation, and testing.

Improved Library Quality and Reliability

LibMake can also help developers improve the quality and reliability of their libraries by providing a set of pre-defined templates that follow best practises. This can help to reduce the number of bugs and errors in libraries, and it can make them more robust and reliable.

Incentives

Enforce Best practises and Generate Initial Code

LibMake can help developers enforce best practises by providing a set of pre-defined templates that follow these practises. It can also generate initial code for common library functionality, which can save developers a significant amount of time.

LibMake offers the following features and benefits:

- Create your Rust library with ease using the command line interface or by providing a configuration file in CSV, JSON, TOML, or YAML format.
- Rapidly generate new library projects with a pre-defined structure and boilerplate code that you can customize with your own template.
- Generate a library pre-defined GitHub Actions workflow to help you automate your library development and testing.
- Automatically generate basic functions, methods, and macros to get you started with your Rust library.

- Enforce best practices and standards with starter documentation, test suites, and benchmark suites that are designed to help you get up and running quickly.

With LibMake, you can easily generate a new Rust library code base structure with all the necessary files, layouts, build configurations, code, tests, benchmarks, documentation, and much more in a matter of seconds.

Try LibMake Today

If you are a developer, I encourage you to try LibMake ⊡ to see how it can help you streamline your library development process. LibMake is a free and an open-source tool, and it is available for download from the GitHub repository ⊡.